

Java XML - Verarbeitung

Folien zur LU Multimedia I
Michael Brandejsky
WS 2004/05

Einführung - XML

- XML - eXtensible Markup Language
 - Sprache zur Darstellung von Informationen
 - Besteht aus Elementen, Attributen und Text :
 - Element : **<Person></Person>**
 - Attribut : **<Person Name="Max"></ Person>**
 - Text : **<Person Name="Max">Informatikstudent</ Person>**
 - Informationen als Subelemente :
 1. **<teilnehmer>**
 2. **<vorname>Max</vorname>**
 3. **<nachname>Musterstudent</nachname>**
 4. **</teilnehmer>**
 - oder als Attribute
 1. **<teilnehmer vorname="Max" nachname="Musterstudent" />**

More Info: <http://www.ims.tuwien.ac.at>

2

Einführung - XML Beispiel

- Beispiel:

```
<?xml version="1.0" encoding="UTF-8"?>
<gruppeninformation>
  <gruppennummer>1b</gruppennummer>
  <teilnehmer vorname="Max" nachname="Musterstudent" />
</gruppeninformation>
```

Elemente : *gruppeninformation, gruppennummer, teilnehmer*

Attribute : *vorname, nachname*

Text : *1b, Max, Musterstudent*

More Info: <http://www.ims.tuwien.ac.at>

3

Einführung - Wichtige Regeln bei XML

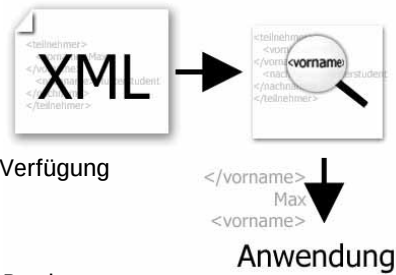
- XML unterscheidet zwischen Groß und Kleinschreibung:
`<person Name="Max"></person>`
`<Person Name="Max"></Person>` ist nicht das selbe!
- Wertzuweisungen bei Attributen nur mit Anführungszeichen:
`<Person Name=Max></person>` falsch
`<Person Name="Max"></Person>` richtig
- Elemente bestehen aus Tags und End-Tags : `<Person></Person>`
 - Ausnahme, leere Elemente:
`<Person>` falsch `<Person/>` richtig
 `<Person />` richtig
- Elementnamen dürfen kleine Umlaute, Ziffern oder Sonderzeichen enthalten

More Info: <http://www.ims.tuwien.ac.at>

4

Einführung - Parser

- **Parser**
Liest die Informationen von XML-Dokumenten aus und stellt sie der Anwendung zur Verfügung
- **Zwei Ansätze dafür:**
 - SAX - Simple API for XML Parsing
 - DOM - Document Object Model

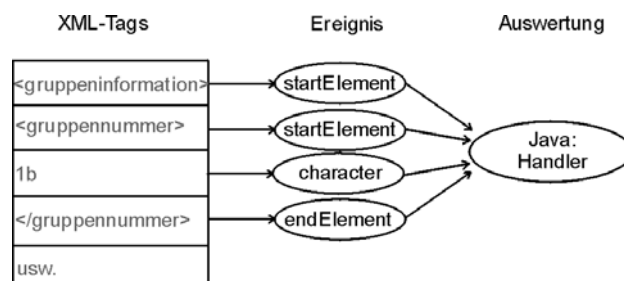


More Info: <http://www.ims.tuwien.ac.at>

5

Einführung - SAX

- **SAX - Simple API for XML Parsing**
Abarbeitung des Dokuments als Strom von Ereignissen (sequenzielles Lesen) mittels Handler



More Info: <http://www.ims.tuwien.ac.at>

6

Einführung - SAX

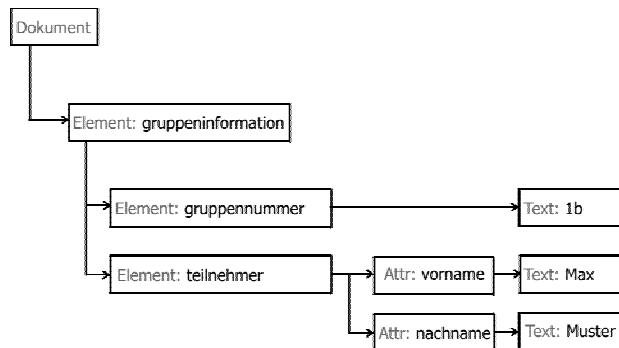
- SAX - Simple API for XML - Parsing
 - Dokument als Strom von Ereignissen
 - **Vorteile**
 - Dateigröße egal, da sich das XML Dokument nie vollständig im Speicher befindet (Bearbeiten von XML Daten ist auch möglich, wenn das XML nicht vollständig in den Speicher passt)
 - Einfach zu benutzen
 - Schnell
 - **Nachteile**
 - Kein freier Zugriff auf beliebige Teile eines Dokumentes ohne Zwischenspeicherung dieser
 - Keine Funktionen zum Schreiben oder Ändern von XML

More Info: <http://www.ims.tuwien.ac.at>

7

Einführung - DOM

- DOM - Document Object Model
 - Das Dokument wird in einer Baumstruktur abgebildet



More Info: <http://www.ims.tuwien.ac.at>

8

Einführung - DOM

- DOM – Document Object Model
 - Hierarchische Struktur (Baum)
 - Wurzel
 - Knoten
 - Blatt
 - **Vorteile**
 - Erzeugen, aktualisieren, lesen und löschen.
 - **Nachteile**
 - Entstehung einer komplexen Baumstruktur
 - Begrenzter Speicherplatz, da die Baumstruktur im Speicher gehalten werden muss
 - Verarbeitungsgeschwindigkeit langsamer

More Info: <http://www.ims.tuwien.ac.at>

9

Einführung – Vergleich SAX/DOM

- SAX - Simple API for XML – Parsing:

Wird nicht gespeichert, kann nicht verändert werden, wird nur eingelesen, kein wahlfreier Zugriff

Verwendet man zum schnellen Auslesen von Daten
- DOM – Document Object Model:

Struktur wird gespeichert, kann verändert werden, ist aber durch den Speicherplatz begrenzt

Verwendet man bei Änderungen der Struktur oder der Daten

More Info: <http://www.ims.tuwien.ac.at>

10

Implementierende APIs

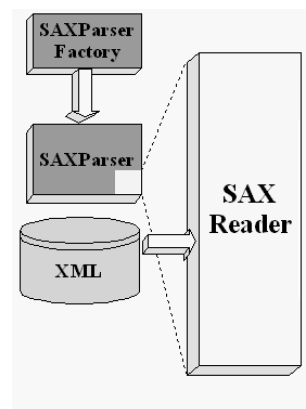
- JAXP - Java API for XML Processing
 - von Sun
 - SAX, DOM
 - Framework unabhängig vom gewählten Parser
 - Factory - Design
- JDOM – Java Document Object Model
 - Nicht von Sun (Brett McLaughlin, Jason Hunter)
 - SAX, DOM
 - Orientiert sich an Java Konzepten
 - Leicht zu erlernen

More Info: <http://www.ims.tuwien.ac.at>

11

JAXP: SAX 2.0 - Parser

- SAXParserFactory
 - Erzeugt Instanz eines Parsers
- SAXParser
 - Definiert Parse()-Methoden
 - Übergabeparameter
 - XML - Datei
 - Handler (Prozedur, welche bestimmt was bei einem bestimmten Ereignis geschieht)
- SAXReader
 - Registrieren der Handler
- Handler
 - DefaultHandler: startElement(), endElement(), characters(), startDocument(), endDocument(), ...



More Info: <http://www.ims.tuwien.ac.at>

12

JAXP: SAX 2.0 – Parser (sax.java)

```

1.  import java.io.FileReader;
2.  import javax.xml.parsers.*;
3.  import org.xml.sax.InputSource;
4.  import org.xml.sax.helpers.DefaultHandler;
5.  public class sax
6.  {
7.      public static void main (String[] args) { myParser(); }
8.      public static void myParser()
9.      {
10.         try {
11.             SAXParserFactory factory = SAXParserFactory.newInstance();
12.             SAXParser parser = factory.newSAXParser();
13.             InputSource is = new InputSource(new FileReader("d:\\lumm1.xml"));
14.             parser.parse(is, new SaxHandler());
15.         }
16.         catch (Exception e) { System.out.println("Fehler: " + e); }
17.     }
18. }

```

JAXP: SAX 2.0 – Parser (SaxHandler.java)

```

1.  import javax.xml.parsers.*;
2.  import org.xml.sax.helpers.*;
3.  import org.xml.sax.*;
4.  public class SaxHandler extends DefaultHandler
5.  {
6.      public void startDocument() throws SAXException
7.      { System.out.println("Document Starts");}
8.      public void endDocument() throws SAXException
9.      { System.out.println("Document End");}
10.     public void characters(char[] ch, int start, int length) throws SAXException
11.     { String charString = new String(ch, start, length);
12.       System.out.println("characters: " + charString); }
13.     public void startElement(String namespaceURI, String localName,
14.                             String qName, Attributes atts) throws SAXException {
15.         System.out.println("startElement: " + qName);
16.         for (int i = 0 ; i < atts.getLength() ; i++) {
17.             System.out.println("Attribute: " + atts.getLocalName(i));
18.             System.out.println("tValue: " + atts.getValue(i)); } }
19.     public void endElement(String namespaceURI, String localName, String qName)
20.     throws SAXException
    { System.out.println("endElement: " + qName); }

```

JAXP: SAX 2.0 – Parser

```

C:\java\xml>java sax
Document Starts
startElement: gruppeninformation
startElement: gruppennummer
characters: 1b
endElement: gruppennummer
startElement: teilnehmer
Attribute:
  Value: Max
Attribute:
  Value: Musterstudent
endElement: teilnehmer
endElement: gruppeninformation
Document End
C:\java\xml>

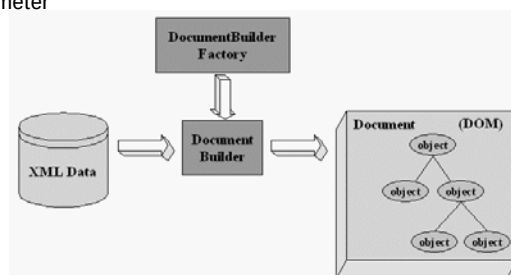
```

More Info: <http://www.ims.tuwien.ac.at>

15

JAXP: DOM - Parser

- DocumentBuilderFactory
 - Erzeugt Instanz eines DocumentBuilder
- DocumentBuilder
 - Erzeugt DOM - Document
 - Übergabeparameter
 - XML - Datei
- Node
 - Document
 - Element
 - Attribute



More Info: <http://www.ims.tuwien.ac.at>

16

JAXP: DOM – Parser (dom.java 1/3)

```

1.  import javax.xml.parsers.*;
2.  import org.xml.sax.InputSource;
3.  import java.io.FileReader;
4.  import java.io.IOException;
5.  import org.w3c.dom.*;

6.  public class dom {
7.      public static void main (String[] args) { myParser(); }
8.      public static void myParser() {
9.          Document document=null;
10.         try {
11.             DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();
12.             DocumentBuilder builder = factory.newDocumentBuilder();
13.             InputSource is = new InputSource(new FileReader("d:\\lumm1.xml"));
14.             document = builder.parse(is);
15.             echo(document.getFirstChild());
16.         }
17.         catch (Exception e)
18.             { System.out.println("Fehler: " + e); }
19.     }

```

JAXP: DOM – Parser (dom.java 2/3)

```

20. public static void echo(Node n)
21. { int type = n.getNodeType();
22.   switch (type)
23.   {case Node.ATTRIBUTE_NODE: System.out.print("ATTR:");
    printlnCommon(n); break;
24.    case Node.DOCUMENT_NODE: System.out.print("DOC:");
    printlnCommon(n); break;
25.    case Node.ELEMENT_NODE: System.out.print("ELEM:"); printlnCommon(n);
26.      NamedNodeMap atts = n.getAttributes();
27.      for (int i = 0; i < atts.getLength(); i++)
28.      { Node att = atts.item(i); echo(att);}
29.      break;
30.    case Node.TEXT_NODE: System.out.print("CHAR:"); printlnCommon(n);
    break;
31.    default: System.out.print("UNSUPPORTED NODE:" + type);
    printlnCommon(n); break;
32.  }
33.  for (Node child = n.getFirstChild(); child != null; child=child.getNextSibling())
34.  { echo(child); }
35. }

```

More Info: <http://www.ims.tuwien.ac.at>

18

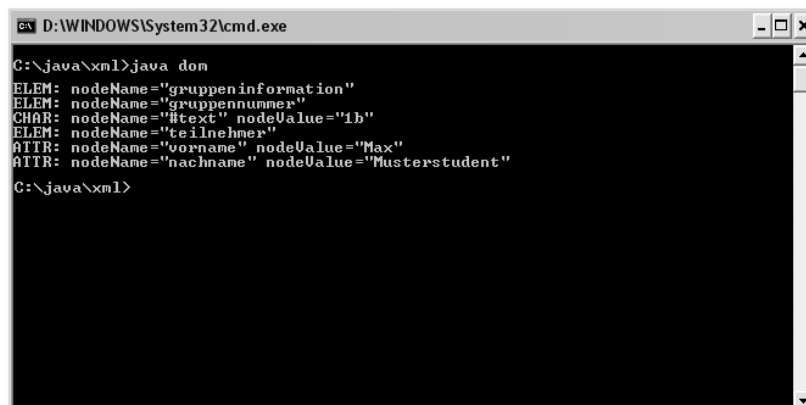
JAXP: DOM – Parser (dom.java 3/3)

```

36. public static void printlnCommon(Node n)
37. {
38.     System.out.print(" nodeName='" + n.getNodeName() + "'");
39.     String val = n.getNodeValue();
40.     if (val != null)
41.     { System.out.print(" nodeValue=");
42.       if (val.trim().equals(""))
43.       {
44.           System.out.print("[WS]");
45.       }
46.       else {
47.           System.out.print("'" + n.getNodeValue() + "'");
48.       }
49.     }
50.     System.out.println();
51. }
52. }

```

JAXP: DOM – Parser



```

D:\WINDOWS\System32\cmd.exe
C:\java\xml>java dom
ELEM: nodeName="gruppeninformation"
ELEM: nodeName="gruppennummer"
CHAR: nodeName="#text" nodeValue="1b"
ELEM: nodeName="teilnehmer"
ATTR: nodeName="vorname" nodeValue="Max"
ATTR: nodeName="nachname" nodeValue="Musterstudent"
C:\java\xml>

```

More Info: <http://www.ims.tuwien.ac.at>

20

JDOM: XML - Datei erzeugen

- Datei:

```
<?xml version="1.0" encoding="UTF-8"?>
<gruppeninformation>
  <gruppennummer>1b</gruppennummer>
  <teilnehmer vorname="Max" nachname="Musterstudent" />
</gruppeninformation>
```

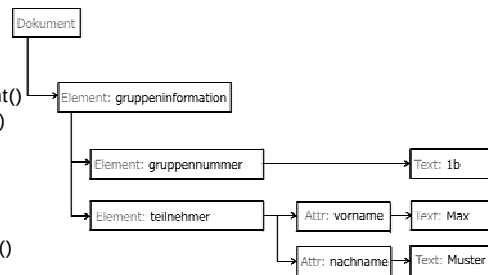
- Schritte

1. Struktur (Baum)

- new Document()
- new Element()
- addContent()
- setAttribute()

2. File erzeugen

- XMLOutputter()



More Info: <http://www.ims.tuwien.ac.at>

21

Beispiel

JDOM: XML - Datei erzeugen (save.java)

```
1. import java.io.*;
2. import org.jdom.*;
3. import org.jdom.output.XMLOutputter;
4. public class save {
5.     public static void main (String[] args) { generateXML(); }
6.     public static void generateXML() {
7.         Document doc = new Document(new Element("gruppeninformation"));
8.         Element gruppennummer = new Element("gruppennummer");
9.         gruppennummer.addContent("1b");
10.        Element teilnehmer = new Element("teilnehmer");
11.        teilnehmer.setAttribute("vorname", "Max");
12.        teilnehmer.setAttribute("nachname", "Musterstudent");
13.        doc.getRootElement().addContent(gruppennummer);
14.        doc.getRootElement().addContent(teilnehmer);
15.
16.        XMLOutputter outputter = new XMLOutputter();
17.        File file = new File("C:\\", "lumm1.xml");
18.        try { file.createNewFile(); FileWriter out = new FileWriter(file);
19.            outputter.output(doc, out); out.close(); }
20.        catch (IOException e) { System.out.println("IO-Fehler: "+e.toString()); }
21.    }
}
```

Sonstiges

- Weiterführende Informationen zu XML & Java
 - <http://java.sun.com/xml/>
 - <http://java.sun.com/webservices/docs/1.0/tutorial/index.html>
 - http://www.galileocomputing.de/openbook/javainsel2/java_130000.htm
- Weiterführende Informationen zu JAXP
 - Download und Installation (Web Services Developer Pack v1.4)
<http://java.sun.com/xml/jaxp/index.jsp>
 - Arbeiten mit XML (SAX, DOM usw.)
<http://java.sun.com/xml/jaxp/dist/1.1/docs/tutorial/TOC.html>
 - API
<http://java.sun.com/webservices/docs/1.4/api/index.html>

More Info: <http://www.ims.tuwien.ac.at>

23

Sonstiges

- Weiterführende Informationen zu JDOM
 - Home
<http://www.jdom.org>
 - Download
<http://www.jdom.org/dist/binary/jdom-1.0.zip>
 - Installation (MS Windows)
 1. z.B.: C:\>set JAVA_HOME=C:\jdk1.3.1_01
 2. z.B.: C:\>set Path=%PATH%;%JAVA_HOME
 3. jdom-1.0.zip entzippen (z.B.: C:\jdom-1.0; siehe README.txt)
 4. z.B.: C:\jdom-1.0\>build.bat
 5. Klassenpfad setzen:
 - a. z.B.: set CLASSPATH=c:\jdom-1.0\build\classes;%CLASSPATH%
 - b. z.B.: set CLASSPATH=c:\jdom-1.0\lib\xerces.jar;c:\jdom-1.0\build\jdom.jar;%CLASSPATH%

More Info: <http://www.ims.tuwien.ac.at>

24